

Esercitazione 7

Nella propria home directory creare una sottodirectory chiamata es07, in cui metteremo tutti i file C di oggi.

Esercizi su funzioni e procedure ricorsive

Quando si ha a che fare con array in funzioni o procedure ricorsive, si possono usare due metodi: il primo utilizza l'aritmetica dei puntatori per aggiornare ogni volta la testa dell'array, il secondo invece si porta dietro un indice per tenere traccia della posizione raggiunta. Ad ogni modo e' sempre necessario portarsi dietro la dimensione (residua o totale, a seconda del caso).

Ricordate che oltre alla funzione o procedura indicata dall'esercizio, si richiede che scriviate anche un main che usi tale funzione e ne dimostri **ogni** funzionalità.

Inoltre, se con l'esercizio viene fornita la firma della funzione/procedura da scrivere, si richiede che la soluzione rispetti questa firma in ogni dettaglio (nome funzione, tipo di ritorno, lista di argomenti).

Per quanto riguarda l'uso dei booleani (true o false) potete usare quanto descritto in seguito:

```
# define FALSE 0;
# define TRUE 1;
```

Esercizi

1) Scrivere una funzione ricorsiva che calcoli il MCD di due numeri interi positivi. Ricordare che la funzione MCD(N,M) termina quando $N == M$ e si ritorna N.

2) Scrivere una funzione che calcoli ricorsivamente il numero di elementi pari di un array passato.

3) Scrivere una procedura ricorsiva che elevi al quadrato tutti gli elementi di un array passato.

4) Scrivere una funzione ricorsiva fattoriale che riceva un intero e ritorni il fattoriale di quel valore.

```
int fattoriale(int val);
```

5) Scrivere una procedura ricorsiva che riceva un array a lo stampi in senso inverso.

```
void stampaInverso1 (int vet[], int dim);
void stampaInverso2 (int vet[], int i, int dim);
```

6) Svolgere l'esercizio precedente stampando sempre e solo il primo elemento dell'array passato (oppure sempre e solo l'ultimo elemento dell'array passato). Suggerimento: che succede se la funzione ricorre prima di stampare?

7) Scrivere un metodo ricorsivo che, dati un carattere c ed un array di caratteri s, restituisce true se c occorre in s, false altrimenti.

8) Scrivere una funzione ricorsiva che calcoli la somma di due numeri senza

usare altra operazione matematica che l'incremento o il decremento di uno.

9) Scrivere una funzione ricorsiva che, assegnati due interi N1 ed N2, restituisca la somma di tutti gli interi compresi tra N1 ed N2.

10) Sia assegnato un vettore A di interi di dimensione N. Scrivere una funzione ricorsiva che calcoli il massimo valore degli elementi di A.

11) Scrivere una funzione ricorsiva che azzeri tutti gli elementi pari, raddoppi gli elementi dispari e ritorni il minimo elemento di un array passato.

12) Scrivere una funzione ricorsiva per la ricerca di un intero all'interno di un vettore di interi passato per parametro. La funzione termina se nel vettore è presente l'elemento ricercato.

13) Scrivere una procedura ricorsiva che inverte la porzione di un array individuata dagli indici from e to.

```
void inverti(int v[], int from, int to);
```

14) La funzione di Ackermann e' una delle piu' semplici funzioni totalmente computabili a non essere ricorsiva primitiva.

http://it.wikipedia.org/wiki/Funzione_di_Ackermann. In pratica, la funzione cresce piu' velocemente di qualsiasi funzione ricorsiva primitiva (compreso qualsiasi esponenziale). La funzione e' definita ricorsivamente per casi (sui naturali):

- $A(m, n) = n + 1$ (se $m = 0$)
- $A(m, n) = A(m - 1, 1)$ (se $m > 0$ e $n = 0$)
- $A(m, n) = A(m - 1, A(m, n - 1))$ (se $m > 0$ e $n > 0$)

Scrivere una funzione che calcoli la funzione di Ackermann.

```
unsigned long Ackermann(unsigned long m, unsigned long n);
```

Quanto vale Ackermann(3,10)? Quanto vale Ackermann(4,1)?

Quanto tempo ci mette a calcolare questi valori?

Avvertenza: non andate oltre questi limiti (soprattutto su m) o l'esecuzione potrebbe non terminare in tempo per il fine settimana...

15) Come visto nell'ultimo esercizio dell'esercitazione 6, una parola si dice 'palindroma' se si può leggere sia da destra che da sinistra. Ad esempio le parole OSSO e INGEGNI sono palindrome. Scrivere una funzione ricorsiva, che preso un array A di caratteri restituisce true se A contiene una parola palindroma.

Esercizi complessi

16) Scrivere una funzione che chieda un valore N all'utente e quindi calcoli ricorsivamente l'N-esimo termine della successione di Fibonacci.
 $F(n) = F(n-1) + F(n-2)$

Riuscite a scrivere la funzione fibonacci ottimizzando (riducendo al minimo) il numero di chiamate ricorsive?

Suggerimento: Per verificare quante chiamate vengono eseguite, usate una variabile globale che viene incrementata ad ogni chiamata.

Suggerimento 2: potete sfruttare il fatto che per calcolare $F(n-1)$ dovete sapere $F(n-2)$?

Suggerimento 3: una funzione in C non può ritornare più di un valore ma

se riceve dei puntatori puo' modificare i valori delle variabili originali.

17) Scrivere una funzione che ricevuto un array di interi, calcoli e stampi ricorsivamente tutte le permutazioni dell'array.